IN THE DRAWINGS:

Please add a PRIOR ART legend above Fig. 1 and Fig. 2.

REMARKS

In the above-identified Office Action, the Examiner rejected Claims 1 - 5, 8 – 12 and 15 – 19 under 35 U.S.C. §112 second paragraph as being indefinite. Claims 1 – 3, 8 – 10 and 15 – 17 were rejected under 35 U.S.C. §103(a) as being unpatentable over P. J. Fleming and J.J. Parker *IBM Technical Disclosure Bulletin, vol. 33, issue 9, Feb 1991* (Fleming here after) in view of Jun Hasegawa et al. (Japanese patent no. JP 61136145). Claims 4 – 7, 11 – 14 and 17 – 20 were indicated as allowable if rewritten in independent form to include all the limitations of the base claim and any intervening claims and to overcome the 112 rejection made thereto.

The Examiner is thanked for the telephone interview of April 5, 2006. In that interview, Claim 1 and one of the references (i.e., the IBM technical Bulletin) were discussed. Specifically, Applicants' attorney mentioned that one of the distinguishing factors of the claimed invention (in relation to the reference) is the provision for data spanning more than one page to be read from storage using only one page fault. The Examiner stated that the reference does provide for all data needed by an application to be cached using a File access history (FAH) value to pre-fetch some, most or all data needed by the application. No agreement was reached.

Applicants have amended the Drawings by adding a PRIOR ART legend above Figs. 1 and 2 as suggested by the Examiner.

Applicants have also amended Independent Claims 1, 8 and 15 to better claim the invention and to include the limitations of Claims 2, 9 and 16, respectively. Consequently, Claims 2, 9 and 16 are canceled and Claims 3, 10 and 17 are amended to change their dependency from a canceled claim to a non-canceled claim.

Further, Claims 1, 4, 5, 8, 11, 12, 15 and 17 – 19 are amended to overcome the 112 rejection made thereto by replacing "memory" with RAM as suggested by the Examiner. New Claim 21 is added for consideration.

By this amendment, Claims 1, 3 – 8, 10 – 15 and 17 – 21 are pending in the Application. For the reasons stated more fully below, Applicants submit that the claims, as amended, are allowable over the applied reference. Hence, reconsideration, allowance and passage to issue are respectfully requested.

As stated in the SPECIFICATION, spatial data pre-fetching is based on the likelihood that once data is referenced, nearby data is also likely to be referenced. That is, the decision to pre-fetch data is determined by the current data block access (e.g., fetching the data block adjacent to the data block currently being accessed). Spatial data pre-fetching works splendidly when data is being read sequentially. For example, after two consecutive page faults of sequentially stored data, a block of sequential pages of data will be pre-fetched through normal data read-ahead. Hence, if future referenced pages are part of the pre-fetched block, which is highly likely when data is being read sequentially, the data will have already been in RAM when needed.

However, if data is being read randomly, spatial data pre-fetching may not work as well. For example, suppose an executing program is randomly reading data from a file. Suppose further that the executing program makes a request to read a certain amount of data that resides on two sequential pages. If the data is not already in RAM, two page faults will be raised in order to load the two pages in the RAM. Because the pages are sequential, the system may infer that data is being read sequentially; and hence, pre-fetch a block of sequential pages of data. Since data is being read randomly, it is highly unlikely that future needed data will be on the pre-fetched block of pages. Thus, the block of pages may have been pre-fetched in vain and the physical pages onto which they are placed wasted. Continually pre-fetching unneeded pages of data may place an undue pressure on RAM space as well as on performance. Thus a need exists for a system and method of improving random multi-page fault-based data pre-fetches.

According to the teachings of the invention, when a piece of data spanning more than one page is to be read from a file that is being read randomly, a

determination will be made as to whether the data is in RAM or in a storage device. If it is determined that the data is in the storage device, the entire piece of data will be read all at once from the storage device using only one page fault.

Since only one page fault is used to read the piece of data (that spans more than one page), data pre-fetches will not occur as data pre-fetching only happens when two or more consecutive page faults are used to read data stored on sequential pages.

The invention is set forth in claims of varying scopes of which Claim 1 is illustrative.

> 1. A method of improving fault-based multi-page pre-fetches comprising the steps of:
> ***determining whether data from a file is being read randomly or sequentially upon receiving a request to read data from the file, the request including a range of data to be read, the range of data spanning more than one page***;
> determining, if data is being read randomly from the file, whether previous data has been read from a random access memory (RAM) or from a storage device;
> ***determining, using the range of data, how much data to read***; and
> ***attempting to read the data from the RAM if previous data has been read from the RAM or read the data all at once from the storage device using only one page fault if previous data has been read from the storage device***. (Emphasis added.)

Applicants submit that the claims are allowable over the applied references.

Fleming teaches a method of using file access history (FAH) to determine the amount of data to pre-fetch. According to the teachings of Fleming, a value is assigned to a file to indicate whether data was read from the file randomly or sequentially. Specifically, when the file is created, it is assigned a default value. The default value indicates sequential access since almost all files are read from beginning to end. After the file is accessed by a program, the value may change

to another value indicating random access, if data was read from the file randomly by the program.

In any case, the FAH value is used by caching and read-ahead programs to determine whether the file was accessed randomly or sequentially. If data was read sequentially from the file, a caching or read-ahead program will pre-fetch as much data as possible to allow for efficient access by the application that is presently reading data from the file. If data was read from the file randomly, the caching or read-ahead program will pre-fetch only enough data to satisfy the application's request.

However, Fleming does not teach the steps of (1) ***determining whether data from a file is being read randomly or sequentially upon receiving a request containing a range of data spanning more that one page to be read from a file*** (Fleming only determines whether data was read randomly or sequentially the last time an application program read data from the file), (2) ***determining, using the range of data, the amount of data to read***; and (3) ***attempting to read the data from the RAM if previous data has been read from the RAM or read the data all at once from the storage device using only one page fault if previous data has been read from the storage device*** as claimed.

Specifically, since enough data, as determined by using the FAH, will be pre-fetched in accordance with the teachings of Fleming, there will not need any reason to determine whether previous data was read randomly or sequentially from the file such that it can be determined whether to ***attempt to read the data from the RAM*** etc. Rather, data will just be read from the file since it is known that the data is already in the RAM. And if, by chance, the data were not already cached in the RAM, then more than one page fault would have to be instituted in order to read all the data needed by the application.

As far as it is understood, Jun purports to teach a method of reducing overhead from an attempt to read data from a cache memory when the data (i.e., instructions in this case) is not in the cache. According to the purported

teachings of Jun, when data is being read, an attempt is made to read the data from the cache. If the data is in the cache (i.e., a cache hit), a value will be stored indicating that a cache hit did occur. If on the other hand, the data is not found in the cache (i.e., a cache miss), another value will be stored indicating that a cache miss did occur. Upon a successive attempt to read data from the cache, this indication will be used to determine whether to concurrently attempt to read the data from memory. Specifically, if it is indicated that a cache hit ensued during the previous attempt to read the data from the cache, there is a high likelihood that the data being read presently is in the cache. Therefore, an attempt to concurrently read the data from regular memory is not needed.

If however, it is indicated that a cache miss did ensue during the previous attempt to read the data from the cache, there is a high likelihood that the data being read presently is not in the cache. Therefore, an attempt to concurrently read the data from regular memory will be instituted. In this case, if the data is indeed not in the cache, the time it took to determine so will not matter as regular memory was concurrently being accessed for the data.

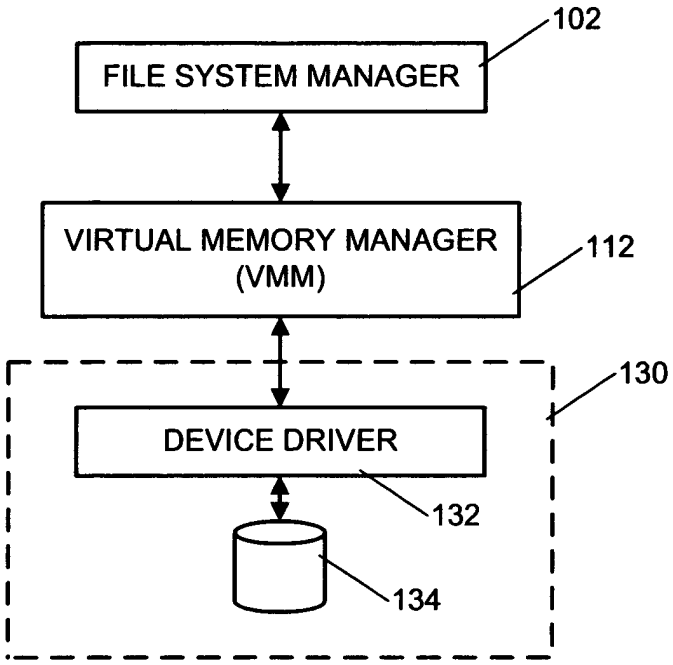
But, just as in the case of Fleming, Jun does not teach the steps of ***using a range spanning more than one page to read the data all at once from the storage device <u>using only one page fault</u> if previous data has been read from the storage device*** when ***data is being read randomly from a file***.

Hence, Applicants submit that Claim 1, as well as its dependent claims, should be allowable. Independent Claims 8 and 15, which all incorporate the above-emboldened-italicized limitations in the above-reproduced claim 1, together with their dependent claims, should also be allowable. Hence, Applicants once more respectfully request reconsideration, allowance and passage to issue of the claims in the application.

Respectfully Submitted

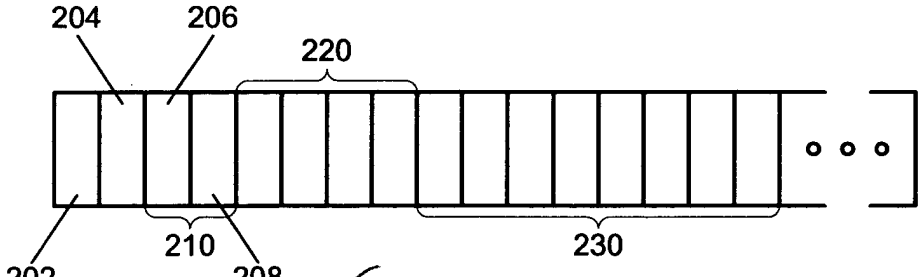
By: _____

Volel Emile
Attorney for Applicants
Registration No. 39,969
(512) 306-7969



(Prior Art)

FIG. 1



(Prior Art)

FIG. 2